



计算机软件与新技术人才培养系列教材

软件技术

数据结构与算法  
C语言程序设计  
C#程序设计  
Python程序设计  
Java程序设计  
Java Web应用开发  
Java框架开发技术  
Web前端开发  
Web应用系统开发  
HTML5与JavaScript程序设计  
HTML5+CSS3网页设计技术  
Android应用开发  
iOS应用开发

大数据技术

Linux操作系统基础  
Spark技术与应用  
Python数据分析  
大数据基础  
数据库高级管理技术  
大数据集群搭建维护与数据存储  
大数据采集与数据处理  
大数据可视化技术应用  
数据挖掘应用  
智能数据分析与应用

云计算技术

云计算基础  
OpenStack系统架构与部署  
云存储规划与部署  
网络存储规划与实施  
云安全技术与应用  
服务器虚拟化技术项目式教程  
虚拟化技术与应用  
云数据中心架构与运维  
公有云运维与应用实践  
云平台建设与维护实战

人工智能技术

人工智能基础  
机器学习  
深度学习  
人工智能算法设计  
人工神经网络技术  
智能视觉技术  
智能算法容器化部署  
自然语言处理技术  
智能语音应用开发

计算机软件与新技术人才培养系列教材

HTML5+CSS3网页设计技术

黄玉春 刘占文 刘 兵 / 主编

中国科学技术出版社

安徽省高水平教材建设项目  
计算机软件与新技术人才培养系列教材

# HTML5+CSS3 网页设计技术

黄玉春 刘占文 刘 兵 / 主编



读科学 得真知  
科学普及出版社漫漫读微信号



定价：56.00元



中国科学技术出版社  
CHINA SCIENCE AND TECHNOLOGY PRESS



安徽省高水平教材建设项目  
计算机科学与新技术人才培养系列教材

# HTML5+CSS3 网页设计技术

---

黄玉春 刘占文 刘 兵 / 主编

中国科学技术出版社  
· 北 京 ·

### 图书在版编目 (CIP) 数据

HTML5+CSS3 网页设计技术 / 黄玉春, 刘占文, 刘兵  
主编. — 北京: 中国科学技术出版社, 2024. 12.  
(计算机软件与新技术人才培养系列教材). — ISBN 978-  
7-5236-1253-8  
I. TP312.8; TP393.092.2  
中国国家版本馆 CIP 数据核字第 2025LJ1979 号

---

策划编辑 王晓义  
责任编辑 杨 洋  
封面设计 唐韵设计  
责任校对 焦 宁  
责任印制 徐 飞

---

出 版 中国科学技术出版社  
发 行 中国科学技术出版社有限公司  
地 址 北京市海淀区中关村南大街 16 号  
邮 编 100081  
发行电话 010-62173865  
传 真 010-62173081  
网 址 <http://www.cspbooks.com.cn>

---

开 本 889 mm × 1194 mm 1/16  
字 数 505 千字  
印 张 18  
版 次 2024 年 12 月第 1 版  
印 次 2024 年 12 月第 1 次印刷  
印 刷 北京荣玉印刷有限公司  
书 号 ISBN 978-7-5236-1253-8/TP · 511  
定 价 56.00 元

---

(凡购买本社图书, 如有缺页、倒页、脱页者, 本社销售中心负责调换)



# 前言

在信息化时代的浪潮中，Web 技术已逐渐成为连接世界、传播知识、推动社会进步的重要工具。作为前端开发的核心，HTML5 和 CSS3 承载着网页内容的展示与布局，它们为网页设计和前端开发提供了强大的工具和可能性。学习 HTML5 和 CSS3，不仅要掌握基础知识，还应深入理解其在实际应用中的作用。

本书响应新时期国家对职业教育的新要求，落实立德树人根本任务，贯彻《高等学校课程思政建设指导纲要》等教育部有关文件精神，将专业知识与思政教育有机结合，实现价值引领、知识传授和能力培养紧密结合，使学生在在学习过程中不仅能够理解技术的原理和应用，更能够领悟其中的道德规范和价值导向。

全书共分 12 个模块。模块一到模块四分别讲解了网页设计概述、HTML5 基础、CSS3 基础和 CSS3 选择器，这是 HTML5 和 CSS3 的基础；模块五讲解了盒子模型，这是网页布局 and 元素美化的基础；模块六到模块八分别讲解了列表与超链接、图像与音视频、表格与表单，这是网页结构设计的基础；模块九讲解了转换、过渡与动画，这是网页前端的特效；模块十和模块十一分别讲解了 CSS 定位技术和网页布局，这是网页设计的布局和排版的方法；模块十二设置了综合实训——制作企业网站，对网站开发的流程进行介绍，并实际开发了“中华文化网”的首页和注册页，便于学生模仿学习。

本书还注重实践性和可操作性，在模块的末尾设置了实战案例和练习题，帮助学生巩固所学知识，提高实际动手能力。同时，本书配有电子教案、教学课件、案例素材及源代码、课后习题及答案等丰富的立体化教学资源，有需要者可发邮件至 2393867076@qq.com 领取。

最终，希望通过对本书的学习，学生不仅能够成为一名优秀的前端开发者，更能够成为一名具有高尚道德情操和强烈社会责任的公民。让我们携手共进，在 HTML5 和 CSS3 的学习道路上不断前行，为构建更加美好的网络世界贡献自己的力量。

本书为安徽省高等学校省级质量工程项目拟立项的教材。本书模块一、模块二、模块八由黄玉春编写，模块三、模块四由巨峰编写，模块五由巨峰、田春尧编写，模块六由吴继兵、吴睿编写，模块七、模块十由吴继兵编写，模块九由黄玉春、刘占文编写，模块十一由刘兵编写，模块十二由吴继兵、邱金峰编写。本书在编写过程中得到了多位同行专家的支持，在此表示感谢。

由于作者水平有限，加之计算机技术发展迅速，书中错误和不足之处在所难免，恳请业内专家、学者和广大读者批评指正。

编 者

2024 年 9 月





# 目录

## 模块一

### 网页设计概述..... 1

#### 任务一 了解网页相关知识 ..... 2

- 一、网站和网页 ..... 2
- 二、动态网页和静态网页 ..... 3
- 三、常用的网页名词 ..... 4
- 四、Web 标准 ..... 5

#### 任务二 网页制作技术入门 ..... 6

- 一、网页标记语言 HTML ..... 6
- 二、网页表现技术 CSS ..... 6

- 三、网页脚本语言 JavaScript ... 6

#### 任务三 网页制作工具 VS Code 的使用 ..... 7

- 一、VS Code 特点 ..... 7
- 二、下载和安装 VS Code ..... 7
- 三、安装 VS Code 插件 ..... 8
- 四、使用 VS Code ..... 10

## 模块二

### HTML5 基础..... 14

#### 任务一 认识 HTML ..... 15

- 一、HTML 的发展历史 ..... 15
- 二、HTML5 的优势 ..... 15

#### 任务二 理解 HTML5 文档结构和基本语法 ..... 16

- 一、HTML5 文档基本结构 ..... 16
- 二、HTML5 基本语法 ..... 17

- 三、HTML5 文档头部相关标签 ... 19

#### 任务三 学习文本标签 ..... 20

- 一、页面格式化标签 ..... 20
- 二、文本格式化标签 ..... 23
- 三、文本语义标签 ..... 24
- 四、特殊字符 ..... 27

#### 任务四 实战案例 ..... 27

## 模块三

### CSS3 基础 ..... 30

#### 任务一 认识 CSS ..... 31

- 一、CSS 的基本概念 ..... 31
- 二、CSS 的特点 ..... 31

#### 任务二 掌握 CSS3 基础选择器 ..... 32

- 一、CSS3 样式规则的定义 ..... 32
- 二、标记选择器 ..... 32
- 三、id 选择器 ..... 32

- 四、类选择器 ..... 33
- 五、通配符选择器 ..... 33
- 六、并集选择器 ..... 33

### 任务三 CSS 的引用 ..... 33

- 一、内部样式表 ..... 33
- 二、外部样式表 ..... 34
- 三、内联样式表 ..... 35
- 四、输入样式表 ..... 36

### 任务四 CSS 的层叠与继承 ..... 37

- 一、CSS 层叠 ..... 37

- 二、CSS 继承 ..... 38
- 三、CSS 继承的应用 ..... 39
- 四、CSS 优先级 ..... 39

### 任务五 CSS 设置文本样式 ..... 40

- 一、CSS 字体样式属性 ..... 41
- 二、CSS 文本外观样式 ..... 43

### 任务六 实战案例 ..... 47

- 一、搭建 HTML 框架 ..... 47
- 二、添加 CSS 样式 ..... 48

## 模块四

# CSS3 选择器 ..... 50

### 任务一 认识 CSS3 组合选择器 ..... 51

- 一、后代选择器 ..... 51
- 二、子元素选择器 ..... 52
- 三、相邻兄弟选择器 ..... 52
- 四、普通兄弟选择器 ..... 53

### 任务二 掌握属性选择器 ..... 54

- 一、属性名选择器 ..... 55
- 二、指定属性值选择器 ..... 55
- 三、属性值包含选择器 ..... 56
- 四、属性值匹配开始值选择器 ..... 58

- 五、属性值匹配结尾值选择器 ..... 59

### 任务三 了解伪类选择器与伪元素 ..... 59

- 一、动态伪类 ..... 60
- 二、结构伪类 ..... 60
- 三、伪元素 ..... 62

### 任务四 学习 CSS 嵌套 ..... 64

### 任务五 实战案例 ..... 66

- 一、搭建 HTML 框架 ..... 66
- 二、添加 CSS 样式 ..... 66

## 模块五

# 盒子模型 ..... 69

### 任务一 认识盒子模型 ..... 70

- 一、盒子模型概述 ..... 70
- 二、盒子大小的计算方式 ..... 70

### 任务二 设置盒子模型边框 ..... 73

- 一、边框的样式 ..... 73
- 二、边框的宽度 ..... 75
- 三、边框的颜色 ..... 77
- 四、CSS3 圆角边框 ..... 79
- 五、CSS3 阴影边框 ..... 81
- 六、CSS3 图片边框 ..... 82

### 任务三 设置盒子模型边距 ..... 83

- 一、CSS 内边距 ..... 83
- 二、CSS 外边距 ..... 85
- 三、外边距合并 ..... 87

### 任务四 设置盒子模型背景 ..... 89

- 一、CSS3 背景 ..... 89
- 二、CSS3 背景透明度 ..... 91
- 三、CSS3 渐变 ..... 93

### 任务五 实战案例 ..... 98

- 一、结构分析 ..... 99



|                |    |
|----------------|----|
| 二、样式分析 .....   | 99 |
| 三、制作页面结构 ..... | 99 |

|                   |    |
|-------------------|----|
| 四、定义 CSS 样式 ..... | 99 |
|-------------------|----|

## 模块六

# 列表与超链接..... 102

### 任务一 掌握列表标签 ..... 103

|               |     |
|---------------|-----|
| 一、无序列表 .....  | 103 |
| 二、有序列表 .....  | 104 |
| 三、自定义列表 ..... | 105 |
| 四、列表的嵌套 ..... | 106 |

### 任务二 CSS 控制列表样式 ..... 107

|                  |     |
|------------------|-----|
| 一、列表符号样式 .....   | 107 |
| 二、图片作为列表符号 ..... | 109 |

|                |     |
|----------------|-----|
| 三、列表符号位置 ..... | 110 |
|----------------|-----|

|                |     |
|----------------|-----|
| 四、列表属性简写 ..... | 111 |
|----------------|-----|

### 任务三 设置超链接 ..... 112

|                     |     |
|---------------------|-----|
| 一、定义超链接 .....       | 112 |
| 二、CSS 伪类控制超链接 ..... | 114 |

### 任务四 实战案例 ..... 116

|                    |     |
|--------------------|-----|
| 一、搭建 HTML 框架 ..... | 116 |
| 二、添加 CSS 样式 .....  | 117 |

## 模块七

# 图像与音视频..... 119

### 任务一 网页中插入图像并定义

#### 样式 ..... 120

|                      |     |
|----------------------|-----|
| 一、常见图像格式 .....       | 120 |
| 二、<img> 标签 .....     | 121 |
| 三、<picture> 标签 ..... | 122 |
| 四、用 CSS 来修饰图像 .....  | 124 |
| 五、图像热点 .....         | 128 |

### 任务二 网页中插入视频与音频 ... 130

|                  |     |
|------------------|-----|
| 一、向网页中插入视频 ..... | 130 |
| 二、向网页中插入音频 ..... | 132 |

### 任务三 实战案例 ..... 133

|                   |     |
|-------------------|-----|
| 一、搭建页面结构 .....    | 134 |
| 二、添加 CSS 样式 ..... | 135 |

## 模块八

# 表格与表单..... 138

### 任务一 表格的使用 ..... 139

|                    |     |
|--------------------|-----|
| 一、创建表格 .....       | 139 |
| 二、单元格跨行跨列表格 .....  | 141 |
| 三、表格列的分组 .....     | 143 |
| 四、CSS 控制表格样式 ..... | 144 |

### 任务二 表单的使用 ..... 146

|               |     |
|---------------|-----|
| 一、表单的构成 ..... | 146 |
| 二、表单控件 .....  | 147 |

|                    |     |
|--------------------|-----|
| 三、CSS 控制表单样式 ..... | 152 |
|--------------------|-----|

### 任务三 实战案例 ..... 155

|                    |     |
|--------------------|-----|
| 一、搭建 HTML 框架 ..... | 155 |
| 二、添加 CSS 样式 .....  | 157 |

## 模块九

## 转换、过渡与动画 ..... 160

## 任务一 CSS3 2D 转换 ..... 161

- 一、转换中的二维坐标系 ..... 161
- 二、2D 位移 ..... 161
- 三、2D 缩放 ..... 163
- 四、2D 旋转 ..... 165
- 五、2D 扭曲 ..... 166
- 六、变换原点 ..... 167
- 七、matrix() 方法 ..... 169

## 任务二 CSS3 3D 转换 ..... 170

- 一、开启 3D 空间 ..... 170
- 二、设置景深 ..... 171
- 三、透视点位置 ..... 171
- 四、3D 旋转 ..... 171
- 五、3D 位移 ..... 173
- 六、3D 缩放 ..... 175

## 任务三 利用 CSS 过渡效果 ..... 176

- 一、过渡属性 ..... 177
- 二、过渡持续时间属性 ..... 178
- 三、过渡速度曲线属性 ..... 178
- 四、过渡延迟时间属性 ..... 180
- 五、过渡复合属性 ..... 180

## 任务四 CSS3 动画制作 ..... 181

- 一、动画中的基本概念 ..... 181
- 二、过渡和动画的区别 ..... 182
- 三、动画的基本使用 ..... 182
- 四、动画的属性 ..... 183

## 任务五 实战案例 ..... 186

- 一、原理分析 ..... 186
- 二、制作页面结构 ..... 187
- 三、定义 CSS 样式 ..... 187
- 四、完整代码 ..... 188

## 模块十

## CSS 定位技术 ..... 192

## 任务一 CSS 定位 ..... 193

- 一、块级元素 ..... 193
- 二、<div> 和 <span> 标签的  
异同 ..... 193
- 三、CSS 定位机制 ..... 194
- 四、相对定位 ..... 194
- 五、绝对定位 ..... 196
- 六、z-index 空间位置 ..... 198

## 任务二 CSS 浮动 ..... 200

- 一、元素的浮动 ..... 200
- 二、清除浮动 ..... 204

## 任务三 弹性盒子 ..... 209

- 一、弹性盒子模型 ..... 209
- 二、弹性盒子常用属性 ..... 209

## 任务四 实战案例 ..... 217

## 模块十一

# 网页布局..... 221

### 任务一 了解 DIV+CSS 排版观念... 222

- 一、将页面用 DIV 分块 ..... 222
- 二、设计各块的位置 ..... 223
- 三、用 CSS 定位 ..... 224

### 任务二 固定宽度且居中的版式... 225

- 一、单列固定宽度居中 ..... 225
- 二、两列固定宽度居中 ..... 226

### 任务三 自适应宽度且居中的版式... 227

- 一、单列自适应宽度且居中  
布局 ..... 228
- 二、两列宽度自适应布局 ..... 229
- 三、左列固定，右列宽度自适应  
布局 ..... 230

- 四、三列浮动中间列宽度自适应  
布局 ..... 231
- 五、圣杯布局 ..... 233

### 任务四 HTML5 语义和结构

#### 元素 ..... 236

### 任务五 实战案例——仿新华网首页

#### (部分) ..... 238

- 一、搭建 HTML 框架 ..... 238
- 二、设置页面样式 ..... 239
- 三、实例完整代码 ..... 240
- 四、采用 DIV+CSS 布局的  
优势 ..... 244

## 模块十二

# 综合实训——制作企业网站..... 247

### 任务一 网站结构规划和页面功能

#### 模块划分 ..... 248

- 一、网站建设主要步骤 ..... 248
- 二、网站结构规划 ..... 248
- 三、网页结构划分 ..... 250
- 四、网页模块命名规范 ..... 251

### 任务二 制作首页 ..... 252

- 一、准备工作 ..... 252
- 二、共用样式 ..... 255
- 三、头部模块制作 ..... 256
- 四、主导航模块制作 ..... 257

- 五、广告模块制作 ..... 258
- 六、主内容模块制作 ..... 260
- 七、友情链接与底部模块制作 ..... 268

### 任务三 制作注册页 ..... 270

- 一、页面结构划分 ..... 270
- 二、头部与底部模块的重用与  
修改 ..... 270
- 三、主体模块制作 ..... 272

# 参考文献..... 276



# 模块一

## 网页设计概述

### 学习目标

#### 知识目标

- (1) 了解网站和网页的概念。
- (2) 了解 HTML、CSS 和 JavaScript 的功能和作用。
- (3) 理解静态网页和动态网页的区别。
- (4) 熟悉网页开发所使用的基本技术和工具。

#### 技能目标

- (1) 能够说出动态网页和静态网页的区别。
- (2) 能够说出网页制作用到的主要技术。
- (3) 能够简单使用网页开发工具 VS Code。

#### 素质目标

- (1) 通过学习网页设计基本知识，掌握网页设计的基本原则和理论，包括页面布局、色彩搭配、图文排版等，具备扎实的网页设计基础知识。
- (2) 通过对网页制作技术的学习，体会 Web 标准的严谨性，从而在工作生活中养成严谨的态度。



# 任务一

## 了解网页相关知识

大家对网页并不陌生,在浏览网上新闻、查询信息、网上购物、看视频等都是在浏览网页。网页可以被看作各种网站应用和信息的容器,所有可视化的内容都会通过网页展示给用户。本任务将讲解网页和网站的基本概念、动态网页和静态网页的区别、常用的网页名词和 Web 标准。

### 一、网站和网页

网站英文为 Web Site。简单来说,网站是多个网页的集合,即根据一定的规则,将用于展示特定内容的相关网页,通过超链接构成一个网站整体。通俗地讲,网站就像互联网上的布告栏一样,人们可以通过网站发布自己想要公开的资讯,或者利用网站提供相关的网络服务。人们可以通过网页浏览器访问网站,获取自己需要的资讯或者享受网络服务,经常浏览的网站有搜狐、新浪、腾讯等。

网页是 Internet 的基本信息单位,英文为 Web Page。网页是以 HTML (HyperText Markup Language, 超文本标记语言) 为基础编写的,能够通过网络传输,并被浏览器翻译成可以显示出来的包含文字、图片、声音、动画等媒体形式的页面文件。进入网站一般情况下,首先看到的是首页,首页集成了指向二级分页以及其他网站的超链接。

例如,中国青年网的首页如图 1-1 所示。



图 1-1 中国青年网首页

网页呈现在用户面前的是各种文字、图像、动画、音频、视频等丰富的内容,网页在本质上是文本文件和其相关的资源,网页最根本的语言是 HTML。HTML 是 Web 编程的基础,是网页设计和开发领域的一个重要组成部分。HTML 指定如何在浏览器中显示网页,它是制作网页的一种标准语言。

通过单击鼠标右键,在弹出的快捷菜单中选择“查看网页源代码”浏览器窗口会显示当前网页的源代码,如图 1-2 所示。

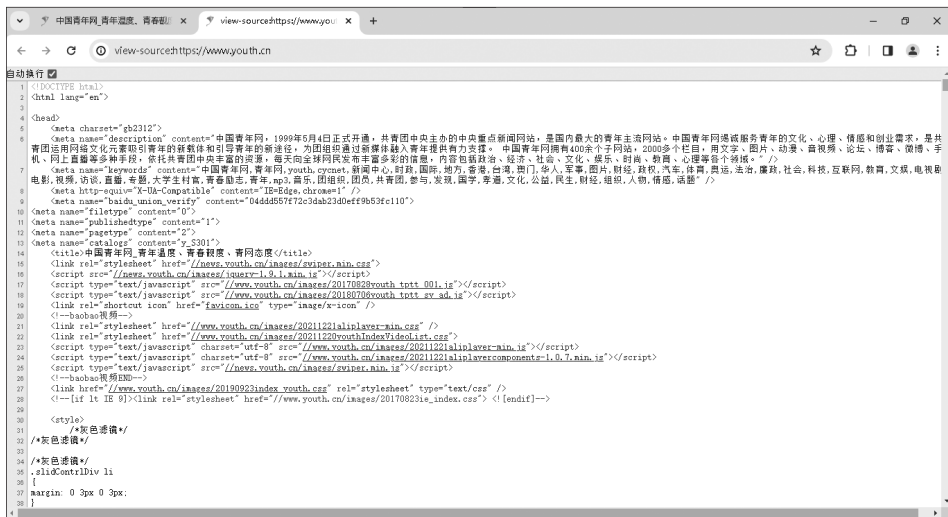


图 1-2 中国青年网首页部分源代码

## 二、动态网页和静态网页

网页有静态和动态之分。静态网页和动态网页的区别不是体现在视觉效果上，而是体现在两者所采用的技术上。

静态网页是指没有后台数据库、不含程序的网页，程序员编写的是什么，它显示的就是什么，不会有任何改变。静态网页更新起来相对比较麻烦，适用于更新较少的展示型网站。静态网页有一个固定的 URL（Uniform Resource Locator，统一资源定位符），且以 .htm、.html、.shtml、.xml 等形式为后缀。发布在服务器上的静态网页是事先保存在服务器上的文件，每个网页都是一个独立的文件，内容相对稳定，容易被搜索引擎检索。

动态网页一般使用 ASP、PHP、JSP、.NET 等网络编程语言编写，是运行于服务器端的代码，浏览时先将服务器端代码执行成 HTML 代码，然后再显示在客户端浏览器中（访问者是无法看到这个文件的源代码的，看到的是通过服务器执行过后的 HTML 代码）。动态网页可以实现的功能较多，如用户注册、登录、在线调查、用户管理、订单管理、站内搜索、即时更新新闻、留言或书写评论等，一般以 .aspx、.jsp、.php 等常见形式为后缀。

### ！ 延伸阅读：Web 前端开发需要学什么？

要成为一名优秀的前端开发人员，需要掌握以下几个方面的技能：

- （1）HTML：学习 HTML 标签（也称“标记”）和语法，能够编写语义化的 HTML 代码，实现页面的结构和内容。
- （2）CSS：学习 CSS 样式和属性，能够编写 CSS 代码，实现页面的布局和样式。
- （3）JavaScript：学习 JavaScript 语言和 DOM 操作，能够编写 JavaScript 代码，实现页面的交互和动态效果。
- （4）前端框架：学习主流的前端框架，如 React、Vue 等，能够使用框架提供的组件和功能，快速开发复杂的单页面应用。
- （5）版本控制：学习 Git 等版本控制工具，掌握代码的管理和维护。
- （6）浏览器调试：学习浏览器的开发者工具，能够进行页面的调试和性能优化。
- （7）网络协议：了解 HTTP、TCP/IP 等网络协议，能够优化页面的加载速度和响应时间。
- （8）Web 安全：掌握常见的 Web 安全漏洞和攻击方式，能够确保网站的安全性和稳定性。

### 三、常用的网页名词

#### 1. IP 地址、域名和 URL

IP 地址是指互联网协议地址（Internet Protocol Address）。IP 地址是 IP 协议提供的一种统一的地址格式，它为互联网上的每一个网络和每一台主机分配一个逻辑地址，以此来屏蔽物理地址的差异。

IP 地址是一组数字，不方便记忆，因此人们为每台计算机赋予了一个具有代表性的名字，这就是主机名。主机名由英文字母或数字组成，将主机名和 IP 对应起来，这就是域名。域名和 IP 地址是可以交替使用的，但一般域名需要通过 DNS 域名解析服务转换成 IP 地址才能找到相应的主机。

URL 是对资源位置的一种表示，是互联网上标准资源的地址。互联网上的每个文件都有一个唯一的 URL，它包含的信息指出文件的位置以及浏览器应该怎么处理它，通常称为 URL 地址。这种地址可以是各种形式的文件，也可以是局域网上的某一台计算机，更多的是 Internet 上的网址。

#### 2. HTTP 和 HTTPS

HTTP（HyperText Transfer Protocol，超文本传输协议）是一种常用的网络通信协议，是客户端浏览器或其他程序与 Web 服务器之间的应用层通信协议。在 Internet 上的 Web 服务器中存放的都是超文本信息，客户机需要通过 HTTP 协议传输所要访问的超文本信息。

HTTPS 是 HTTP 的加密版本。它通过结合 HTTP 和安全套接字层（SSL）或传输层安全协议（TLS）来加密 Web 服务器和客户端之间传输的数据。HTTPS 加密用于保护敏感数据，如密码、信用卡号和其他个人信息。

HTTP 协议传输的数据都是未加密的，也就是明文的，所以使用 HTTP 协议传输隐私信息非常不安全，为了保证隐私信息的安全传输。网景公司设计了 SSL 协议进行 HTTP 协议传输数据的加密，由此诞生了 HTTPS。

HTTP 和 HTTPS 的区别主要在于以下几点：

（1）加密。加密是 HTTP 和 HTTPS 之间的主要区别之一。HTTPS 使用 SSL 或 TLS 来加密数据，使其比 HTTP 安全得多。当通过 HTTPS 传输时，数据在通过 Internet 发送之前被加密。这种加密有助于防止未经授权访问敏感数据，例如信用卡号和密码。

（2）证书认证。证书认证是 HTTP 和 HTTPS 的另一个区别。当 Web 浏览器通过 HTTPS 连接到 Web 服务器时，服务器会向浏览器发送数字证书。该证书包含特定于服务器的信息，包括服务器的公钥。浏览器随后使用此证书与服务器建立安全连接。

（3）端口号。HTTP 使用端口 80，而 HTTPS 使用端口 443。这意味着通过 HTTP 访问网站时，URL 以开头 http://，而通过 HTTPS 的 URL 以开头 https://。

（4）表现。由于加密和解密数据的额外开销，HTTPS 通常比 HTTP 慢。然而，SSL 和 TLS 协议的进步显著降低了这种开销，使 HTTPS 比以前快得多。

#### 3. WWW 和 W3C 组织

WWW（World Wide Web）中文译文“万维网”。WWW 是基于客户机 / 服务器方式的信息发现技术和超文本技术的综合。它不是网络，也不代表 Internet，它只是 Internet 提供了一种服务——网页浏览服务。

W3C 组织是对网络标准制定的一个非营利组织，W3C 是 World Wide Web Consortium（万维网联盟）的缩写，像 HTML、XHTML、CSS、XML 的标准就是由 W3C 来定制。

## 4. 浏览器

浏览器是查看网页的一种工具，它可向服务器发送各种请求，并对从服务器发来的超文本信息和各种多媒体数据格式进行解释、显示和播放。用户需要在计算机上安装浏览器来显示网页中的信息，这是使用互联网的基本条件之一。一般来说操作系统中都已经内置了浏览器，例如 Windows 操作系统内置有微软公司的 Edge 浏览器。用户也可以自行安装浏览器，常用的浏览器有 Google Chrome 浏览器、360 安全浏览器、腾讯 QQ 浏览器等。不同浏览器对网页的解析可能存在差异，本书案例主要使用 Google Chrome 浏览器显示网页效果。

## 四、Web 标准

Web 标准不是某一个标准，而是一系列标准的集合。在 Web 标准中，规定网页主要由 3 部分组成：结构（Structure）、表现（Presentation）和行为（Behavior）。对应的标准也分 3 方面：结构化标准语言，主要包括 HTML、XML 和 XHTML；表现标准语言，主要包括 CSS；行为标准，主要包括对象模型（如 W3C DOM）、ECMAScript 等。这些标准大部分由万维网联盟起草和发布，也有一些是其他标准组织制定的标准，比如 ECMA（European Computer Manufacturers Association，欧洲计算机制造商协会）的 ECMAScript 标准。

### 1. 结构

结构对网页中用到的信息进行整理和分类。在结构中用到的主要技术包括 HTML、XML 和 XHTML。

HTML 超文本标记语言是 Web 的基本描述语言。HTML 是由 HTML 命令组成的描述性文本，HTML 命令可以说明文字、图形、动画、声音、表格、链接等。HTML 的结构包括头部（Head）、主体（Body）两大部分。头部描述浏览器所需的信息，主要包含所要说明的具体内容。目前，HTML 最新版本是 HTML5。

XML（eXtensible Markup Language）是一种可扩展标记语言。XML 最初设计的目的是弥补 HTML 的不足，具有强大的扩展性满足网络信息发布需要，后来逐渐用于网络数据的转换和描述。

XHTML（eXtensible HyperText Markup Language）可扩展超文本标记语言。是在 HTML4.0 的基础上，用 XML 的规则对其进行扩展，得到了 XHTML。简单地说，建立 XHTML 的目的是实现 HTML 向 XML 的过渡。目前，已逐渐被 HTML5 所取代。

### 2. 表现

表现技术用于对已被结构化的信息进行显示上的控制，包含版式、颜色、大小等形式控制。在网页制作中主要 CSS 设置网页的样式。

CSS（Cascading Style Sheets）中文译为“层叠样式表”。W3C 创建 CSS 标准的目的是希望以 CSS 来描述整个页面的布局设计，与 HTML 所负责的结构分开。目前，CSS 的最新版本是 CSS3。

### 3. 行为

行为是指对整个文档内部的一个模型定义及交互行为的编写，用于编写用户可进行交互式操作的文档。表现行为的 WEB 标准技术主要有：DOM（Document Object Model）文档对象模型和 ECMAScript（JavaScript 的扩展脚本语言）两部分。

根据 W3C DOM 规范，DOM 是一种让浏览器与内容结构之间沟通的接口，使用户可以访问页面上的标准组件。ECMAScript 脚本语言用于实现具体的界面上对象的交互操作。



## 任务二

# 网页制作技术入门

HTML、CSS 和 JavaScript 是网页制作的标准语言，要学好网页制作技术，首先要对它们有总体的认识。本节任务介绍网页标记语言 HTML、网页样式设计所使用的技术 CSS、最常用的脚本语言 JavaScript 等。

### 一、网页标记语言 HTML

HTML 是用于创建网页和设计其他可在网页浏览器中看到的信息的一种标记语言，它以纯文字格式为基础。可以使用任何文字编辑器或所见即所得的 HTML 编辑器来编辑 HTML 文件。HTML 被用来结构化信息——例如标题、段落、列表和图像等，主要负责网页的“内容”部分。如果想要专业地学习网页的设计和编辑，必须具备一定的 HTML 知识。掌握好 HTML 是精通网页制作的最根本要求。本书讲解的 HTML 语言是最新的 HTML5 版本。

### 二、网页表现技术 CSS

CSS 是 Cascading Style Sheet 的缩写，中文译为“层叠样式表”或“串联样式表”，简称“样式表”。W3C 创建 CSS 标准的目的是以 CSS 取代 HTML 表格式布局、帧和其他表现的语言，用来定义网页外观样式，特别是进行网页的排版布局。HTML 和 CSS 分别实现了网页内容和样式的设计，实现了结构和外观的分离，使站点的访问及维护更加容易。

如今网页排版愈发复杂，布局样式都需要通过 CSS 来实现。采用 CSS 技术可以方便有效地对页面进行布局，更加精确地控制网页的字体、颜色、背景和其他效果。内容相同的网页，只需要对 CSS 样式进行一些改变，就可以实现不同的页面外观和格式。学好 CSS 技术是精通网页制作的基本要求。

CSS 最新版本是 CSS3。在 CSS3 中增加了很多新样式，例如圆角效果、块阴影与文字阴影、使用变形处理（旋转、缩放、倾斜、移动）等，这些内容将在后续的讲解中介绍。

### 三、网页脚本语言 JavaScript

脚本语言由 ASCII 码构成，是一种不必事先编译，只要利用适当的解释器就可以执行的简单程序。在网页中使用脚本语言，可以丰富网页的表现力，是网页设计中很重要的一种技术。目前常用的脚本语言有 JavaScript、VBScript 和 JScript，其中 JavaScript 是众多网页开发者首选的脚本语言。JavaScript 是一种属于网络的脚本语言，已经被广泛用于 Web 应用开发，常用来为网页添加各式各样的动态功能，为用户提供更流畅美观的浏览效果。通常 JavaScript 代码可以直接嵌入 HTML 文件中，随网页一起传送到客户端浏览器，然后通过浏览器来解释执行。

## 任务三

# 网页制作工具 VS Code 的使用

Visual Studio Code (简称 VS Code) 是微软公司推出的一款免费、开源的代码编辑器, 一经推出就受到开发者的欢迎。对于前端开发者来说, 一个强大的编辑器可以使开发者变得简单、便捷、高效。可以用来编写网页前端的编辑器软件有很多, 例如记事本、Notepad++、Dreamweaver、HBuilder、WebStorm、atom 等。本书选择 Visual Studio Code 编辑器作为开发工具。

### 一、VS Code 特点

VS Code 在功能上做到了够用, 体验上做到了好用, 更在拥有海量插件的情况下做到了简洁流畅。具有如下主要优点:

(1) 轻量级。在保证功能强大的同时, VS Code 的文件大小和占用内存较小, 启动速度快。

(2) 跨平台。不同的开发人员为了工作需要, 会选择不同的平台来进行项目开发, 这在一定程度上限制了编辑器的使用范围。VS Code 支持 Windows、macOS 和 Linux 等多个操作系统, 而且使用起来也非常简单。

(3) 智能感知。提供了先进的智能感知功能, 可以自动补全代码、提示关键字等, 并减少开发者出错的可能性。其内置的 IntelliSense 技术还可以根据编程语言、项目类型等因素提供不同的智能感知选项。

(4) 多语言支持。VS Code 支持多种编程语言, 包括 JavaScript、TypeScript、Python、C# 等。它提供了相应的插件和工具, 让开发者可以在同一个编辑器中完成多种任务。

(5) 提供丰富的插件。VS Code 的插件扩展系统是其最重要的特点之一, 其中包含大量实用的插件, 覆盖了几几乎所有的开发场景。

### 二、下载和安装 VS Code

打开浏览器, 登录 VS Code 官方网站。在网站的首页可以看到软件下载链接按钮, 如图 1-3 所示。

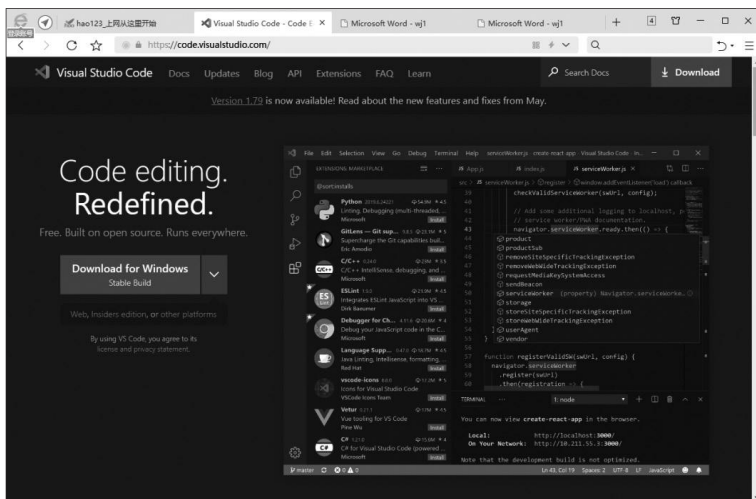


图 1-3 VS Code 官方网站



单击“Download for Windows”按钮，该页面会自动识别当前的操作系统并下载相应的安装包。如果需要下载其他系统的安装包，可以单击按钮右侧的小箭头，在弹出的菜单列表中就会看到其他系统版本的下载列表。

下载完成后，双击安装包启动安装程序，然后按照程序的提示一步一步进行操作，直到完成安装。

### 三、安装 VS Code 插件

VS Code 编辑器提供了非常丰富的插件功能，根据开发的需要，安装对应的插件，可以大大提高开发效率。本节介绍本课程需要用到的 VS Code 常用插件。

#### 1. Chinese ( Simplified )

VS Code 下载下来默认是英文的，可以下载 Chinese (Simplified) Language Pack for Visual Studio Code 插件将编译器汉化，提高开发效率。其安装步骤如下：

(1) 启动 VS Code，单击左侧的“扩展”按钮，打开“扩展商店”。

(2) 在“扩展商店”搜索框中输入关键字“Chinese”，即可找到“Chinese (Simplified) Language Pack for Visual Studio Code”这个插件，如图 1-4 所示。

(3) 单击该插件右下角的“Install”按钮进行安装。也可以单击插件详情页的“Install”按钮进行安装。



图 1-4 安装 VS Code 插件

(4) 安装完成后，在右下角弹出的对话框“In order to use VS Code Simplified, VS Code needs to start.”中，单击“Restart”，重新启动后，VS Code 就变成了简体中文界面了，如图 1-5 所示。



图 1-5 VS Code 中文界面

## 2. Auto Rename Tag

该插件的功能是自动修改标签名，重命名一个开始标签时，自动重命名配对的结束标签。例如，当用户重命名一个 HTML/XML 标签时，该标签将自动重命名所有成对的 HTML/XML 标签。Auto Rename Tag 插件如图 1-6 所示。

所有插件的安装方法都是一样的，后续不再赘述。



图 1-6 Auto Rename Tag 插件

## 3. Open in browser

安装完以后在目标的 HTML 文件上右击鼠标，在弹出的菜单上选择“open in default browser”选项，即可使用默认浏览器预览当前文件；或者使用快捷键“Alt+B”在浏览器打开当前文件。open in browser 插件如图 1-7 所示。

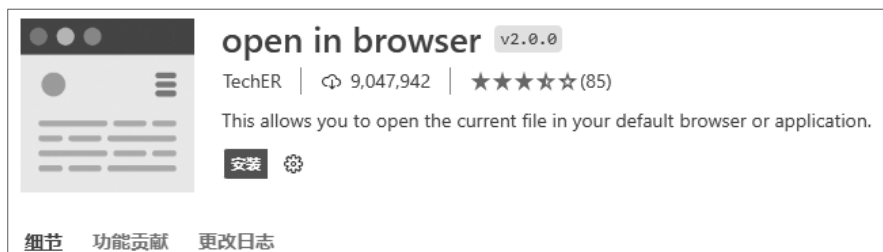


图 1-7 Open in browser 插件

## 4. Live Server

该插件功能是开启一个实时的本地服务器。这是一个非常好用的插件，每次保存文件后到浏览器都要刷新才能看到最新的变化，有了这个插件，就可以实时监听文件的变化，自动刷新网页。Live Server 插件如图 1-8 所示。



图 1-8 Live Server 插件

## 5. One Dark Pro

VS Code 有一些自带的颜色主题，用户可以根据自己的喜好选择相应的颜色主题。One Dark Pro 是最受欢迎的暗黑主题，安装完成后，用户就可以选择使用该主题。

其方法如下：

(1) 单击右下角的齿轮形状  “管理”按钮，在弹出的菜单中选择“颜色主题”，打开主题列表。也可通过快捷键【Ctrl+K+T】直接打开“颜色主题”列表。

(2) 在打开的主题列表中，选择“One Dark Pro”主题，如图 1-9 所示。



图 1-9 选择 VS Code 主题

## 四、使用 VS Code

VS Code 编辑器和常用插件安装完成后，就可以使用 VS Code 制作和编辑网页文件了。

### 1. VS Code 界面

启动 VS Code 编辑器，主界面如图 1-10 所示。编辑器最左侧黑色背景的是“活动栏”，在活动栏上有“资源管理器”“搜索”“代码管理器”“运行和调试”“扩展”“账户”和“管理”等按钮。

单击活动栏上的按钮可以打开相应的功能。例如：单击编辑器左侧的活动栏的“扩展”按钮，可以看到 VS Code 安装的插件；单击活动栏上的“资源管理器”按钮，显示或隐藏“资源管理器”区域。在资源管理器中，可以看到当前项目包含的文件夹和文件。



图 1-10 VS Code 编辑器主界面

在 VS Code 编辑器“开始”界面中，单击“打开文件夹……”，也可以使用快捷键“Ctrl+K+O”直接打开“打开文件夹”对话框，选择某个文件夹作为项目的根目录。在弹出“是否信任此文件夹中的文件作者？”对话框中，单击“是，我信任此作者”按钮；否则，其中的代码可能无法运行，如图 1-11 所示。

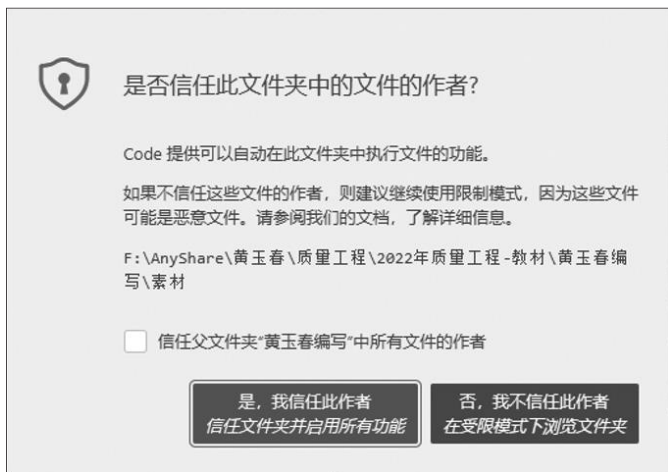


图 1-11 选择信任打开的文件夹

## 2. 在 VS Code 中创建文件

在编辑器的“资源管理器”中，单击新建文件按钮，创建一个文件，命名为“index.html”，如图 1-12 所示。

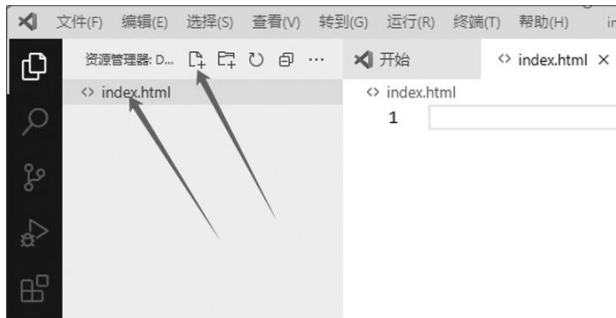


图 1-12 新建 HTML 文档

单击这个文件，进入代码编辑环境。在代码编辑器环境，用户可以编写代码了。代码写好后，可以单击活动栏上的“运行和调试”按钮，运行编写的代码。

在 VS Code 代码编辑器中编辑 HTML 文档，可以通过输入英文感叹号“!”，按【Tab】键或【Enter】键，快速生成 HTML 结构模版代码。

对于 HTML 文档，可以在当前文件窗口右击鼠标，在弹出的快捷菜单上选择“Open with Live Server”“Open in Default Browser”和“Open in Other Browsers”三个选项之一，在浏览中查看网页设计的效果，如图 1-13 所示。

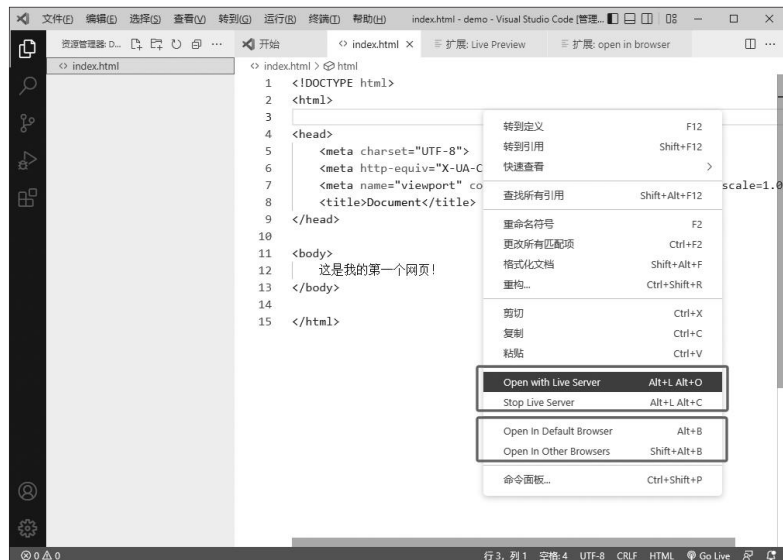


图 1-13 运行网页代码选项

说明：VS Code 安装了“Live Server”插件，才会有“Open with Live Server”选项。安装了“Open in Browser”插件，才会有“Open in Default Browser”和“Open in Other Browser”选项。

选择“Open with Live Server”选项时，会打开系统默认浏览器查看网页效果，当网页代码修改时，浏览器自动更新网页最新效果，用户不需要刷新浏览器。

选择“Open in Default Browser”选项时，会打开系统默认浏览器查看网页效果，当网页代码修改时，需要刷新浏览器才能看到修改后的网页效果。

选择“Open in Other Browser”选项时，会弹出浏览器列表，用户可以选择一个浏览器来查看网页效果。当网页代码修改时，用户也需要刷新浏览器才能看到修改后的网页效果。

## 课后练习

### 一、判断题

1. 因为静态网页的访问速度快，所以现在互联网上的所有网站都是静态网页组成的。( )
2. “HTTP”是一种详细规定了浏览器和万维网服务器之间互相通信的规则。( )
3. 在 Web 标准中，表现是指网页展示给访问者的外在样式。( )
4. 在网页中，层叠样式表指的是 JavaScript。( )
5. 所有的浏览器对同一个 CSS 样式的解析都相同，因此页面在不同浏览器下的显示效果完全一样。( )



## 二、单项选择题

1. 关于 Web 标准的描述, 下列说法正确的是 ( )。  
A. Web 标准只包括 HTML 标准  
B. Web 标准是由浏览器的各大厂商联合制定的  
C. Web 标准并不是某一个标准, 而是一系列标准的集合  
D. Web 标准主要包括结构标准、表现标准和行为标准 3 个方面
2. 关于 HTML 的描述, 下列说法正确的是 ( )。  
A. HTML 是更严谨纯净的 XHTML 版本  
B. HTML 提供了许多标签, 用于对网页内容进行描述  
C. 目前最新的 HTML 版本是 HTML5  
D. 初期的 HTML 在语法上很宽松
3. 关于 CSS 的描述, 下列说法正确的是 ( )。  
A. 当 CSS 作为独立的文件时, 必须以 .html 为后缀名  
B. CSS 用于设置 HTML 页面中的文本内容、图片的外形以及版面的布局等外观显示样式  
C. 只有独立的 CSS 文件才符合结构与表现分离的特点  
D. 目前流行的 CSS 版本为 CSS3
4. HTTP 的中文含义是 ( )。  
A. 文件传输协议  
B. 超文本传输协议  
C. 顶级域名网址  
D. 以上都不是
5. 构成 Web 站点的最基本的单位是 ( )。  
A. 网站  
B. 主页  
C. 网页  
D. 文字

## 三、填空题

1. 网页的基本元素包括\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_和动画等。
2. 根据使用技术的不同, 网页分为\_\_\_\_\_网页和\_\_\_\_\_网页。
3. WWW (World Wide Web) 中文译文是\_\_\_\_\_。WWW 是基于\_\_\_\_\_方式的信息发现技术和超文本技术的综合。
4. 在 Web 标准中, 规定网页主要由 3 部分组成, 分别是\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。
5. 静态网页常见的后缀名有\_\_\_\_\_和\_\_\_\_\_。

## 四、简答题

1. 简要描述 HTTP 和 HTTPS 的差异。
2. 简述什么是 CSS 以及 CSS 的作用。
3. 简述什么是 Web 标准。

## 五、操作题

1. 下载安装 VS Code, 并安装常用的插件。
2. 配置 VS Code。编辑器主题、文档自动保存、编辑器字体字号、文档格式化等。
3. 用 VS Code 编写一个简单的 HTML 网页, 在浏览器中预览这个网页, 显示“HELLO WORLD”字样。

## 模块二

# HTML5 基础

### 学习目标 >

#### ➔ 知识目标

- (1) 了解 HTML 的发展历史。
- (2) 了解 HTML5 的优势。
- (3) 掌握 HTML5 文档结构和基本语法。
- (4) 掌握 HTML5 文本标签。

#### ➔ 技能目标

- (1) 能够写出 HTML5 文档结构。
- (2) 能够用文本标签对页面元素进行标识。

#### ➔ 素质目标

- (1) 通过学习 HTML5 基础知识，培养实践能力和解决问题的能力。
- (2) 通过学习 HTML5 文档结构，体会缩进格式对代码逻辑分析的重要性，在代码编写的过程中养成良好的习惯，提高学习效率。

# 任务一

## 认识 HTML

打开一个网页，查看它的源代码，会看到一些有规律的英文代码。这些代码就是超文本标记语言。“超文本”就是指页面内可以包含图片、链接甚至音乐、程序等非文字的元素，“标记”就是说它不是程序语言，只是由文字及标记符号组合而成。

一个网页无论看上去多么五花八门、生动活泼，其实最本质的东西，就是由这些看着十分单调的 HTML 元素组成的，浏览器或者其他可以浏览网页的设备将这些 HTML 元素“翻译”过来，并按照定义的格式显示出来，转化成最终看到的网页。本任务将介绍 HTML 发展历史和 HTML5 的优势。

### 一、HTML 的发展历史

现在常常习惯于用数字来描述 HTML 的版本（如：HTML5），但是最初并没有 HTML1，而是 1993 年 IETF 团队的一个草案，并不是成型的标准。

两年之后，在 1995 年 HTML 有了第二版，即 HTML 2.0，当时是作为 RFC1866 发布的。

有了以上的两个历史版本，HTML 的发展可谓突飞猛进。1996 年 HTML3.2 成为 W3C 推荐标准。之后在 1997 年和 1999 年，作为升级版本的 4.0 和 4.01 也相继成为 W3C 的推荐标准。

HTML 4.01 发布之后，很长一段时间里都能很好地满足广大互联网用户的日常需求。随着时间的推移，人们开始在网上听歌、玩游戏、看视频，但是 HTML 所支持的网页中并不能直接插入视频、音频、动画等，这个时候 Adobe 率先抓住商机，将 Flash 作为浏览器插件满足了上述需求。

为了推动 Web 标准化运动的发展，在 2004 年，由 Opera、Mozilla 基金会和苹果这些浏览器厂商联合成立了一个叫作 Web Hypertext Application Technology Working Group（Web 超文本应用技术工作组，简称 WHATWG）的组织。WHATWG 专注于 Web 表单和应用程序，而 W3C 专注于 XHTML 2.0。

在 2006 年，双方决定进行合作，创建一个新版本的 HTML，并正式将新的 HTML 命名为“HTML5”。2008 年 1 月 22 日第一份 HTML5 正式草案公布。2014 年 10 月 29 日，W3C 宣布，经过接近 8 年的艰苦努力，该标准规范终于制定完成。

### 二、HTML5 的优势

从 HTML 4.0、XHTML 到 HTML5，从某种意义上讲，这是 HTML 描述性标记语言的一种更加规范的过程。因此 HTML5 并没有给开发者带来多大的冲击，但 HTML5 增加了很多非常实用的新功能和特性。HTML5 的优势如下。

#### 1. 解决了跨平台问题

HTML5 最显著的优势在于跨平台性，用 HTML5 搭建的站点与应用可以兼容 PC 端与移动端、Windows 与 Linux、安卓与 IOS。它可以轻易地移植到各种不同的开放平台、应用平台上，打破各自为政的局面。这种强大的兼容性可以显著地降低开发与运营成本，可以让企业特别是创业者获得更多的发展机遇。

#### 2. 本地存储特性

HTML5 的本地存储特性也给使用者带来了更多便利。基于 HTML5 开发的轻应用比本地 App

拥有更短的启动时间、更快的联网速度，而且无须下载占用存储空间，特别适合手机等移动媒体。

### 3. 更多的多媒体元素

HTML5 让开发者无须依赖第三方浏览器插件即可创建高级图形、版式、动画以及过渡效果，这也使得用户用较少的流量就可以欣赏到炫酷的视觉听觉效果。

### 4. 代码更安全

使用 HTML5，代码更安全。众所周知，Web 应用有一个很大的问题就是代码安全问题，现在 HTML5 可以将 Web 代码全部加密，本地应用解密后再运行，极大地保证了代码的安全性。

### 5. 化繁为简的优势

作为当下流行的通用标记语言，HTML5 尽可能地简化，严格遵循了“简单至上”的原则，主要体现在这几个方面：

- (1) 新的简化的字符集声明。
- (2) 新的简化的 DOCTYPE。
- (3) 简单而强大的 HTML5 API（应用程序接口）。
- (4) 以浏览器原生能力替代复杂的 JavaScript 代码。

为了实现这些简化操作，HTML5 规范需要比以前更加细致、精确。为了避免造成误解，HTML5 对每一个细节都有着非常明确的规范说明，不允许有任何的歧义和模糊出现。

## 任务二

## 理解 HTML5 文档结构和基本语法

学习任何一门语言，首先要掌握该门语言的基本格式和基本语法。HTML5 是一种纯文本语言，也就是说，HTML5 代码在运行时不用事先编译为二进制代码，而是直接通过网页浏览器逐行解释执行。所以，用一般的文本编辑器就可以编写 HTML5 代码，保存时只需要把代码文件扩展名保存为 .htm 或 .html 格式即可。本任务将介绍 HTML5 的基本结构和基本语法。

### 一、HTML5 文档基本结构

在使用 VS Code 编辑 HTML5 文档时，在代码编辑器中输入“!”（半角），接着按下 Tab 键，VS Code 编辑器会自动生成 HTML5 的文档结构，如图 2-1 所示。

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Document</title>
8   </head>
9   <body>
10
11 </body>
12 </html>
```

图 2-1 展示了 HTML5 文档的基本结构。代码中，第 1 行是文档类型声明（<!DOCTYPE html>），第 2 行是文档根标签（<html lang="en">）。第 3 行到第 8 行是文档头部标签（<head>），包含字符集声明（<meta charset="UTF-8">）、兼容性声明（<meta http-equiv="X-UA-Compatible" content="IE=edge">）、视口声明（<meta name="viewport" content="width=device-width, initial-scale=1.0">）和标题（<title>Document</title>）。第 9 行到第 11 行是文档主体标签（<body>），第 12 行是文档结束标签（</html>）。

图 2-1 HTML5 的文档结构

图 2-1 所示 HTML5 结构代码中，`<!DOCTYPE>` 为文档类型声明，它和 `<html>`、`<head>`、`<body>` 共同组成了 HTML5 的文档结构，具体介绍如下。

### 1. `<!DOCTYPE>`

`<!DOCTYPE>` 位于文档最前面，用来声明文档类型。是用来告知浏览器用哪种规范来解释这文档中的代码的；DOCTYPE 语句必须在文档的第一行。HTML5 文档中可以不使用 DOCTYPE，浏览器也可以勉强识别，但是建议要有 DOCTYPE 语句。

### 2. `<html>`

`<html>` 位于 `<!DOCTYPE>` 之后，也被称为根标签。根标签主要用于告知浏览器自身是一个 HTML 文档，其中 `<html>` 标志 HTML 文档的开始，`</html>` 标志 HTML 文档的结束，在它们之间是文档的头部和主体内容。标签内的“`lang= 'en'`”表示此页面定义为英文网页，也可以将其改为“`lang=zh`”（zh 表示中文），如果写的不是英文网站，最好把这段去掉。

### 3. `<head>`

`<head>` 标签用于定义 HTML 文档的头部信息，也被称为头部标签，紧跟在 `<html>` 标签之后。头部标签主要用来封装其他位于头部的标签，例如，在网页的头部可以用 `<title></title>` 标记来定义网页的标题，可以用 `<meta>` 标记定义与文档相关的信息，可以放置 JavaScript 代码块或其他定义部分。

### 4. `<body>`

`<body>` 标签用于定义 HTML 文档所要显示的内容，也被称为主体标签。浏览器中所有显示的文本、图像、表格、超链接、音频和视频等信息都必须放置在 `<body>` 标签内，才能最终展示给用户。

注意：

一个 HTML 文档只有一对 `<html>` 标签、一对 `<head>` 标签和一对 `<body>` 标签，且 `<head>` 和 `<body>` 都必须在 `<html>` 之内，`<head>` 和 `<body>` 是并列关系，`<body>` 位于 `<head>` 之后。

## 二、HTML5 基本语法

从 HTML 文档的基本结构可以看出，HTML 文档是由按照一定规则组合起来的各种标记组成的。前面介绍了 `<html>`、`<head>` 和 `<body>` 标签都是 HTML 文档的基本标签，除了这些标签，HTML 文档还有大量的标签。下面对标签及标签中的主要属性进行简要的介绍。

### 1. 什么是标签

用尖括号“`<`”和“`>`”括起来的部分称为标记（也称标签），每个标记都必须有一个标记名称来作为该标记的唯一标识，如 `<html>` 中的“html”。绝大部分标记都有其相关的属性及属性值，如 `<body bgcolor="#cccccc">`，其中“`bgcolor`”是标记 `<body>` 的一个属性，“`#cccccc`”是“`bgcolor`”的取值。取值可以用引号括起来，也可以不用引号括起来，标记的属性通常都有一个默认值，如“`bgcolor`”的默认值是“`#ffff`”。



## 2. 标签的分类

根据标签的组成特点，通常将标签分为两大类，分别是双标签和单标签。

(1) 双标签是指由开始和结束两个标签符号组成的标签。一般格式如下：

```
< 标签名 > 内容 </ 标签名 >
```

例如，前面介绍的 `<html>` 和 `</html>`、`<body>` 和 `</body>` 等都是双标签。

(2) 单标签是指用一个标签符号即可完整地描述某个功能的标签，一般格式如下：

```
< 标签名 />
```

在上面的语法格式中，标签名和“/”之间有一个空格，该空格在 HTML 语法中可以省略。

在 HTML 文档中还有一种特殊的单标签——注释标签。其作用主要是帮助程序员阅读和理解 HTML 文档。一般格式如下：

```
<!-- 注释内容 -->
```

注释内容不会显示在浏览器窗口中，但是作为 HTML 文档内容的一部分，注释内容可以被下载到用户的计算机上，用户查看源代码时也可以看到注释内容。

### ! 延伸阅读：单标签的由来

HTML 标签的作用原理就是选择网页内容进行描述，也就是说，需要描述哪些内容，就选择该内容，通过双标签定义内容的开始和结束。但单标签本身就可以描述一个功能，不需要定义内容的开始和结束。例如，水平线 `<hr>` 标签按照双标签的语法规则，它应该写成“`<hr></hr>`”，但是水平线标签不需要选择网页内容，本身就代表一条水平线，此时如果写成双标签，代码就显得有些冗余。这就需要用单标签来简化代码结构。

## 3. 标签的嵌套与并列

在 HTML 文档中会存在很多标签，各标签之间都具有一定的关系。标签的关系主要有嵌套关系和并列关系两种。

嵌套关系也称为包含关系，可以简单理解为一个双标签里面包含其他的标签。例如在 HTML5 结构代码中，`<html>` 标签和 `<head>` 标签与 `<body>` 标签就是嵌套关系。

```
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

需要注意的是，在标签的嵌套过程中，必须先结束最靠近内容的标签，再按照由内到外的顺序依次关闭标签。下面的代码显示了标签嵌套的正确与错误写法对比。

```
<p><i><strong> 中国特色社会主义 </strong></i></p>
<!-- 正确的嵌套 -->
<p><i><strong> 中国特色社会主义 </i></strong></p>
<!-- 错误的嵌套 -->
```

在嵌套关系的标签中，通常把最外层的标签称为“父标签”，里面的标签称为“子标签”。

标签的并列关系也称为兄弟关系，是指两个标签处于同一级别，并且没有包含关系。例如，在 HTML 的文档结构代码中，`<head>` 标签和 `<body>` 标签就是并列关系。

## 4. 标签的属性

很多标记还有自己的属性，利用这些属性，可以作进一步的详细设置。其语法格式为：

```
<标签名 name1="value1" name2="value2" ...>...</标签名>
```

各属性项间用空格分隔，属性值可用双引号或单引号，也可以不用引号。

例如，下面代码设置宽度 200px，居中对齐的一条线。

```
<hr width="200px" align="center">
```

在 HTML 开始标签中，可以通过“属性 = 属性值”的方式为标签添加属性，其中“属性”和“属性值”就是以键值对的形式出现的。

所谓“键值对”可以理解为对“属性”设置“属性值”。键值对有多种表现形式，如 color=“red”、width=200px 等，其中 color 和 width 即为“键值对”中的“键”（key），red 和 200px 为“键值对”中的“值”（value）。

键值对广泛应用于编程中，HTML 属性的定义形式“属性 = 属性值”只是键值对中的一种。

提示：

值得一提的是，在实际开发中不提倡使用 HTML 标签属性控制元素的样式，推荐使用 CSS 样式表控制元素的属性。关于 CSS 样式将在后续任务中进行介绍。

## 三、HTML5 文档头部相关标签

文档的头部描述了文档的各种属性和信息，包括文档的标题、在 Web 中的位置以及和其他文档的关系等。绝大多数文档头部包含的数据都不会真正作为内容显示在浏览器中。

<head> 标签用于定义文档的头部，它是所有头部元素的容器。<head> 中的元素可以引用脚本、指示浏览器在哪里找到样式表、提供元信息等。

### 1. 设置页面标题标签

<title> 标签是用来描述网页标题的，在浏览器中显示的位置是在浏览器窗口的标题栏上。例如，将页面标题设置为“中国青年网”，具体代码如下：

```
<title> 中国青年网 </title>
```

<title> 标签是 <head> 标签中唯一必须要求包含的东西，<head> 标签内一定要写 <title>。<title> 的增加有利于 SEO 优化。SEO 是搜索引擎优化的英文缩写。通过对网站内容调整，满足搜索引擎的排名需求。

### 2. 定义页面元信息标签

<meta> 标签用来描述一个 HTML 网页文档的元信息，可重复出现在 <head> 头部标签中。<meta> 是一个单标签，本身不包含任何内容，仅仅表示网页的相关信息。

在 HTML5 中，<meta> 标签通过 3 组属性定义页面的相关参数。

1) <meta charset="值"/>

该组属性用来定义文档的字符编码。例如，将字符集设置为 UTF-8 的代码如下：

```
<meta charset="UTF-8">
```

2) <meta http-equiv="名称" content="值">

在 <meta> 标签中使用 http-equiv 和 content 可以设置服务器发给浏览器的 HTTP 头部信息，为

浏览器显示该页面提供相关的参数标准。

`<meta http-equiv="Refresh" content="n;url=http://yourlink">` 定时让网页在指定的时间  $n$  内, 跳转到指定的页面。

例如, 下面的代码可以实现页面停留 30 秒后跳转到新浪网。

```
<meta http-equiv="Refresh" content="30;url=http://www.sina.com.cn">
```

3) `<meta name="名称" content="值">`

在 `meta` 标签中使用 `name` 和 `content` 属性可以为搜索引擎提供信息, 其中 `name` 属性用于提供搜索内容的名称, `content` 属性提供对应的搜索内容值, 具体应用如下:

(1) 设置网页关键字 `Keywords` (关键字)。

说明: 为搜索引擎提供的关键字列表。

用法: `<meta name="Keywords" content="关键词 1, 关键词 2, 关键词 3, 关键词 4, …">`

注意:

各关键词间用英文逗号, 隔开。META 的通常用处是指定搜索引擎用来提高搜索质量的关键字。

(2) 设置网页描述 `Description` (简介)。

说明: `Description` 用来告诉搜索引擎网站主要内容。

用法: `<meta name="Description" content="网页的简述">`

(3) 设置网页作者 `author` (作者)。

说明: 标注网页的作者或制作组

用法: `<meta name="Author" content="张三, abc@sina.com">`

注意:

`content` 可以是作者或制作组的名字, 或 email。

(4) 设置视图模式 `viewport` (视图模式)。

例如, 下面的代码设置视图模式为:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

## 任务三

### 学习文本标签

无论页面内容如何丰富, 文本始终都是页面中的基本元素。为了使文本排版整齐、结构清晰, HTML 中提供了一系列文本控制标签, 如标题标签 `<h1> ~ <h6>`, 段落标签 `<p>`, 换行标签 `<br />` 等。本节任务将对文本控制标签进行详细讲解。

#### 一、页面格式化标签

一篇结构清晰的文档通常都会通过标题、段落、分割线等进行结构排列, HTML 网页也提供了相应的格式化标签, 下面进行介绍。

## 1. 标题标签

标题（heading）标签有 6 个级别，从 <h1> 到 <h6>。<h1> 为最大的标题，<h6> 为最小的标题。通过设定不同等级的标题，可以完成很多层次结构的设置，比如文档的目录结构或者一份写作大纲。该标签的语法格式为：

```
<hn align=" 对齐方式 "> 标题文本 </hn>
```

align 属性的值有以下 3 种。

- （1）left：左对齐（默认值）。
- （2）center：居中对齐。
- （3）right：右对齐。

【例 2-1】标题标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> 标题标记的使用 </title>
</head>
<body>
  <h1> 一级标题 </h1>
  <h2> 二级标题 </h2>
  <h3> 三级标题 </h3>
  <h4> 四级标题 </h4>
  <h5> 五级标题 </h5>
  <h6> 六级标题 </h6>
</body>
</html>
```

文件在浏览器中的预览效果如图 2-2 所示。



图 2-2 设置标题标签

由图 2-2 可以看出，标题默认是左对齐的。可以通过设置 align 属性值，改变标题对齐方式。

## 2. 段落标签

段落（paragraphs）标签 <p> 是处理文字时经常用到的标签。段落内也可以包含其他标签，如图片标签 <img>、换行符标签 <br>、链接标签 <a> 等。该标签的语法格式为：

```
<p align=" 对齐方式 "> 这是一个段落 </p>
```

align 属性的值与标题标签的属性值一样。

【例 2-2】段落标记的使用。

```
<!DOCTYPE html>
<html>
```

```
<head>
  <meta charset="UTF-8">
  <title>段落标记的使用 </title>
</head>
<body>
  <h3 align="center">段落标记的使用 </h3>
  <p align="center">高举中国特色社会主义伟大旗帜 为全面建设社会主义现代化国家而团结奋斗 (居中对齐) </p>
  <p>高举中国特色社会主义伟大旗帜 为全面建设社会主义现代化国家而团结奋斗 (默认左对齐) </p>
</body>
</html>
```

文件在浏览器中预览的效果如图 2-3 所示。

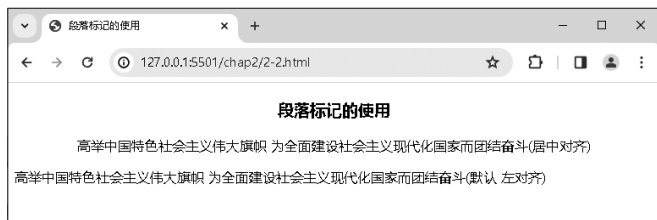


图 2-3 段落标记

### 3. 换行标签

换行标签 `<br />` 是一个单标签，也就是说，它只有起始标签，而没有结束标签。当需要结束一行，并且不想开始新段落时，使用 `<br />` 标签。`<br />` 标签不管放在什么位置，都能够强制换行。

【例 2-3】换行标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>换行符的使用 </title>
</head>
<body>
  <h3 align="center">换行符的使用 </h3>
  <p>这是一个段落，该段落内插了换行符。<br />这个文字跟上面是同一个段落的，是被换行符强制换行了！ </p>
  <p>这是另一个段落 (注意换行符与段落另起一行的区别) </p>
</body>
</html>
```

文件在浏览器中预览的效果如图 2-4 所示。



图 2-4 换行符的使用

### 4. 水平线标签

在网页中经常看到一些水平线将标题与段落，或者段落与段落之间隔开，使文档结构清晰、层次分明。水平线可以通过 `<hr />` 标签来定义，基本语法格式如下。



```
<hr 属性 = “属性值” />
```

`<hr>` 是单标签，在网页中添加一个 `<hr>` 就会显示一条水平线。可以通过为 `<hr>` 标签设置属性和属性值来改变水平线的样式。常用的属性如下：

- (1) `align`：设置水平线的对齐方式。`left` 左对齐（默认值），`right` 右对齐，`center` 居中对齐。
- (2) `size`：设置水平线的粗细。默认为 2 像素。
- (3) `color`：设置水平线颜色。可用颜色名称、十六进制 `#RGB` 表示。
- (4) `width`：设置水平线宽度。可以用像素值，也可以用百分比（默认值 100%）。

【例 2-4】水平线标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> 水平线的使用 </title>
</head>
<body>
  <h1> 神舟十七号载人飞船发射取得圆满成功 </h1>
  <hr />
  <p> 新华社酒泉 10 月 26 日电（记者李国利、黄一宸、刘艺）10 月 26 日 11 时 14 分，搭载神舟十七号载人飞船的长征二号 F 遥十七运载火箭在酒泉卫星发射中心点火发射，约 10 分钟后，神舟十七号载人飞船与火箭成功分离，进入预定轨道，航天员乘组状态良好，发射取得圆满成功。
  </p>
</body>
</html>
```

文件在浏览器中预览的效果如图 2-5 所示。



图 2-5 水平线标记的使用

## 二、文本格式化标签

在网页中，有时需要将文本设置粗体、倾斜或下划线等一些特殊显示的文本效果，为此，HTML 提供了专门的文本格式化标签，使文字以特殊的方式显示。常用的格式化标签如表 2-1 所示。

表 2-1 HTML 文本格式化标签

| 标签                                                   | 描述           |
|------------------------------------------------------|--------------|
| <code>&lt;b&gt;</code> 和 <code>&lt;strong&gt;</code> | 定义粗体文本（加重语气） |
| <code>&lt;i&gt;</code> 和 <code>&lt;em&gt;</code>     | 定义斜体字（着重文字）  |
| <code>&lt;small&gt;</code>                           | 定义小号字        |
| <code>&lt;sub&gt;</code>                             | 定义下标字        |
| <code>&lt;sup&gt;</code>                             | 定义上标字        |
| <code>&lt;u&gt;</code> 和 <code>&lt;ins&gt;</code>    | 定义下划线（插入字）   |
| <code>&lt;s&gt;</code> 和 <code>&lt;del&gt;</code>    | 定义删除字        |

### 【例 2-5】文本格式化标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> 文本格式化标签的使用 </title>
</head>
<body>
  <h3 align="center"> 文本格式化标签的使用 </h3>
  <p><b> 这个文本是加粗的 (使用了 b 标签) </b></p>
  <p><strong> 这个文本是加粗的 (使用了 strong 标签) </strong></p>
  <p><em> 这个文本是斜体的 (使用了 em 标签) </em></p>
  <p><i> 这个文本是斜体的 (使用了 i 标签) </i></p>
  <p><small> 这个文本是缩小的 </small></p>
  <p><del> 这个文本带有删除线 (使用了 del 标签) </del></p>
  <p><s> 这个文本带有删除线 (使用了 s 标签) </s></p>
  <p><u> 这个文本带有下划线 </u></p>
  <p><ins> 这个文本带有下划线 </ins></p>
  <p>a<sup>2</sup>+bx+c=0</p>
  <p>x<sub>1</sub>=?x<sub>2</sub>=?</p>
</body>
</html>
```

文件在浏览器中的预览效果如图 2-6 所示。



图 2-6 文本格式化标签的使用

## 三、文本语义标签

文本语义标签主要用于向浏览器和开发者描述标签的意义，是一些供机器识别的标签，访问者只能看到显示样式的差异。有些文本语义标签可以突出文本的层次关系，方便搜索引擎搜索，甚至提高浏览器的解析速度。在 HTML5 中，文本语义标签有很多，下面介绍 time 标签。

### 1. <time>

<time> 标签用于定义时间和日期，可以代表 24 小时中的某一时间。time 标签不会在浏览器中呈现任何特殊效果，但该标签能够以机器可读的方式对日期和时间进行编码，用户能够将生日提醒或其他事件添加到日程表中，搜索引擎也能够生成更智能的搜索结果。time 有以下两个属性。

(1) datetime：用于定义相应的时间或日期。取值为具体时间或具体日期，不定义该属性时，由文本的内容给定时间和日期。

(2) `pubdate`: 用于定义 `time` 标签中的文本 (或 `article`) 发布日期。

【例 2-6】`time` 标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>time 标签的使用 </title>
</head>
<body>
  <p id="p1">
    <time datetime="2024-4-9"> 今天是 2023 年 4 月 9 日 </time>
  <p>
  <p id="p2">
    <time datetime="2024-4-9T21:00"> 现在时间是 2023 年 4 月 9 日晚上 9 点 </time>
  <p>
  <p id="p3">
    <time datetime="2024-5-1" pubdate="true"> 本消息发布于 2024 年 5 月 1 日 </time>
  </p>
</body>
</html>
```

该例中, `<p>` 元素 `id` 号为 “p1” 中的 `<time>` 元素表示的是日期。页面在解析时, 获取的是属性 “`datetime`” 中的值, 而标记之间的内容只是用于显示在页面中。`<p>` 元素 `id` 号为 “p2” 中的 `<time>` 元素表示的是日期和时间, 它们之间使用字母 “T” 进行分隔。如果在整个日期与时间的后面加上一个字母 Z, 则表示获取的是 UTC (世界统一时间) 格式。`<p>` 元素 `id` 号为 “p3” 中的 `<time>` 元素表示的是发布日期。在 `<time>` 元素中增加了 “`pubdate`” 属性, 表示此日期为发布日期。属性 “`pubdate`” 表示时间是否为发布日期, 它是一个布尔值, 该属性不仅可以用于 `<time>` 元素, 还可用于 `<article>` 元素。

文件在浏览器中的预览效果如图 2-7 所示。



图 2-7 `time` 标签的使用

## 2. `<mark>`

`<mark>` 元素主要功能是在文本中高亮显示某个或某几个字符, 旨在引起用户的特别注意。其使用方法与 `<em>` 和 `<strong>` 有相似之处, 但相比而言, HTML5 中新增的 `<mark>` 元素在突出显示时, 更加随意与灵活。

【例 2-7】`mark` 标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>mark 标签的使用 </title>
</head>
<body>
```

```
<h3>优秀开发人员的<mark>素质</mark></h3>
<p>一个优秀的 web 页面开发人员，必须具有<mark>过硬</mark>的技术与<mark>务实</mark>的专业精神!
</p>
</body>
</html>
```

文件在浏览器中的预览效果如图 2-8 所示。



图 2-8 mark 标签的使用

从图 2-8 中可以看出，使用 <mark> 标签处理字符“素质”“过硬”和“务实”高亮显示。

### 3. <pre>

<pre> 标签可定义预格式化的文本。被包围在 <pre> 标签元素中的文本通常会保留空格和换行符，而文本也会呈现为等宽字体。

提示：

<pre> 标签的一个常见应用是用来表示计算机的源代码。

【例 2-8】<pre> 标签的使用。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>pre 标签的使用</title>
</head>
<body>
  <pre>
    #include <stdio.h>;
    int main()
    {
        /* 我的第一个 C 程序 */
        printf("Hello, World! \n");
        return 0;
    }
  </pre>
</body>
</html>
```

文件在浏览器中的运行效果如图 2-9 所示。

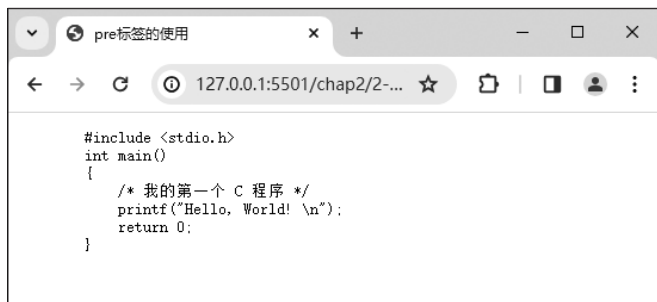


图 2-9 pre 标签的使用

从图 2-9 中可以看出, 包含在 `<pre>` 标签内的文本原样输出。

## 四、特殊字符

在网页上经常看到一些包含特殊字符的文本, 如数学公式、版权信息等。这些特殊的文本 HTML 中提供了专门的替代代码, 如表 2-2 所示。

表 2-2 常用的特殊符号标签

| HTML 原代码                  | 显示结果               | 描述          |
|---------------------------|--------------------|-------------|
| <code>&amp;lt;</code>     | <code>&lt;</code>  | 小于号或显示标记    |
| <code>&amp;gt;</code>     | <code>&gt;</code>  | 大于号或显示标记    |
| <code>&amp;amp;</code>    | <code>&amp;</code> | 可用于显示其他特殊字符 |
| <code>&amp;quot;</code>   | <code>"</code>     | 引号          |
| <code>&amp;reg;</code>    | <code>®</code>     | 已注册         |
| <code>&amp;copy;</code>   | <code>©</code>     | 版权          |
| <code>&amp;trade;</code>  | <code>™</code>     | 商标          |
| <code>&amp;nbsp;</code>   |                    | 不断行的空白      |
| <code>&amp;plusmn;</code> | <code>±</code>     | 正负号         |
| <code>&amp;divide;</code> | <code>÷</code>     | 除号          |
| <code>&amp;times;</code>  | <code>×</code>     | 乘号          |
| <code>&amp;yen;</code>    | <code>¥</code>     | 人民币符号       |

## 任务四

### 实战案例

编排一个新闻页面, 参考格式如图 2-10 所示。



图 2-10 实战案例参考图

观察页面结构, 新闻主要包括新闻主标题、新闻来源、新闻内容、新闻页脚等信息。可以通过定义 HTML 的标题标签、段落标签等来搭建网页的结构。HTML 页面结构代码如下。

```
<!DOCTYPE html>
<html>
<head>
```





2. HTML5 文档结构只能有一对 <html> 标签, 但根据需要可以有多对 <body> 标签。( )
3. <body> 标签和 <head> 标签是并列关系。( )
4. 设置标签属性时, 标签名与属性、属性与属性之间均以空格分开。( )
5. 绝对路径就是网页上的文件或目录在硬盘上的真正路径。( )

## 二、单项选择题

1. 下列选项中, 属于 HTML5 扩展名的是 ( )。  
A. .xhtml      B. .html      C. .htm      D. .xhtm
2. 下列选项中, 可以设置文字字体的属性是 ( )。  
A. face      B. size      C. color      D. font
3. 以下标记符中, 用于设置页面标题的是 ( )。  
A. <title>      B. <caption>      C. <head>      D. <html>
4. 以下标记符中, 属于单标记的是 ( )。  
A. <body>      B. <br>      C. <html>      D. <title>
5. 下列标签中, 不属于文本语义标签的是 ( )。  
A. time      B. pre      C. p      D. mark

## 三、填空题

1. 根据标签的组成特点, 通常将标签分为两大类, 分别是\_\_\_\_\_和\_\_\_\_\_。
2. 在文档的结构关系中, 通常将标签分为\_\_\_\_\_关系和\_\_\_\_\_关系两种。
3. 描述一个 HTML 网页文档元信息的标签是\_\_\_\_\_, 通过设置它的\_\_\_\_\_属性可以定义文档的字符编码。
4. 标题标签一共有\_\_\_\_\_级, 其中字号最大的标题标签是\_\_\_\_\_, 最小的是\_\_\_\_\_。
5. 段落标签是\_\_\_\_\_, 它的\_\_\_\_\_属性用来设置文本的水平对齐方式。

## 四、简答题

1. 请简要描述一下 HTML5 的优势。
2. 请简要描述一下 cite 标签的作用。

## 五、操作题

1. 调试教材中的各个示例。
2. 编写一个网页, 效果参考图 2-12。



图 2-12 操作题效果